

Verilog Debug Tips

Verilog易犯錯誤與訣竅

- 變數跟常數有關的四則運算，都要標明常數的bit，才不會有無法預期的錯誤。
- 變數名稱打錯並不會跳error，會被當成新的wire，訊號會恆0。
- reg要在always block內；wire要在外面用assign。
- switch case要有default；if statement要有else。
- 多bit常數可以用直接寫、串接、複製表示。
 - 直接寫 4'b1111 或 4'd15
 - 串接 {2'b11, 2'b11}
 - 複製 {4{1'b1}}

Module Instantiation

- 可透過名稱或順序來接線，推薦以名稱接線較好檢查

```
module ALU(  
    .src1_i( rs1 ),  
    .src2_i( mux_alusrc ),  
    .ctrl_i( alu_ctrl ),  
    .result_o( alu_result ),  
    .zero_o( zero )  
);
```

※別忘了加“.”

```
module ALU(  
    rs1,  
    mux_alusrc,  
    alu_ctrl,  
    alu_result,  
    zero  
);
```

Verilog的cout: \$display

- 可將訊號印出來看
- Debug時可調整%b, %o, %d, %x，以方便閱讀
- `$display("Reg x1: %0d", cpu.RF.Reg_File[1]);`

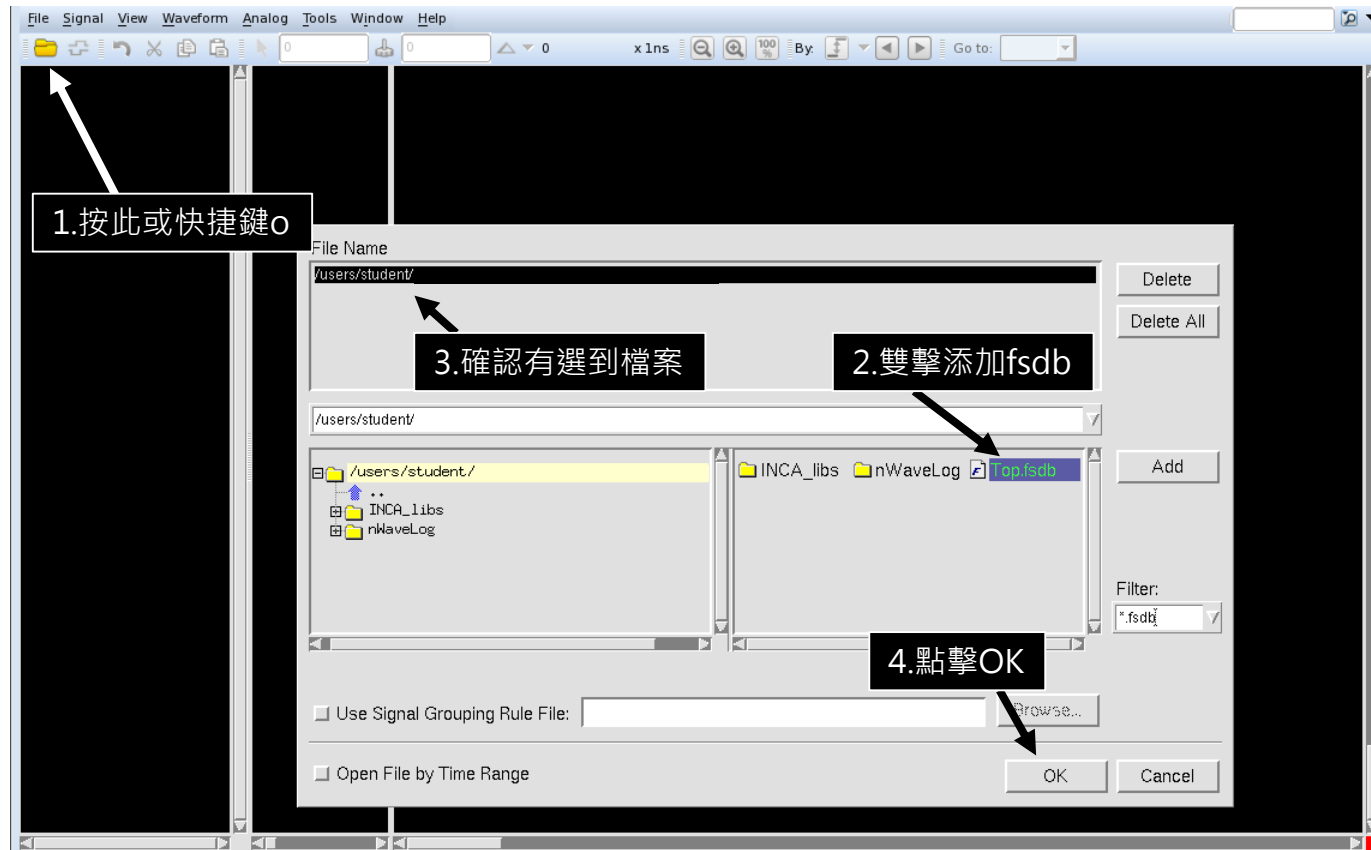
產生Waveform

- Testbench在initial block內加上
 - `$fsdbDumpfile("Top.fsdb");` `//waveform file`
 - `$fsdbDumpvars(0, "+mda");` `//also dump 2D register`
- Makefile要加上`+access+r`
- `ncverilog Simulator.v Test_Bench.v +access+r`
- 使用nWave看waveform

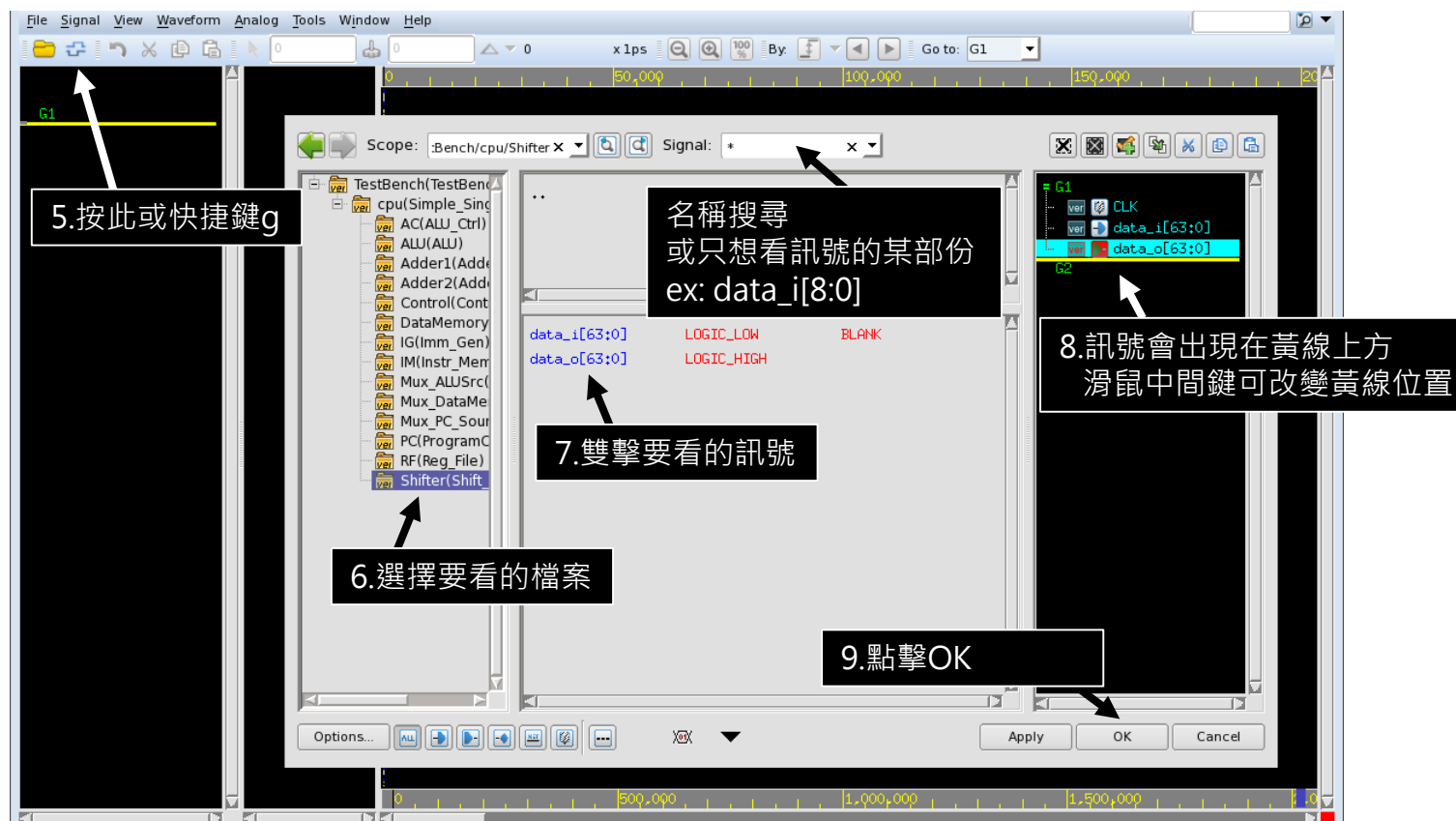
nWave開啟

- 在登入工作站的特定一台machine時，一定要加“ -X ”
- ssh -X ic23
- 開啟nWave時，可加 “&” 將指令置於背景，就不用卡一個頁面
- nWave &

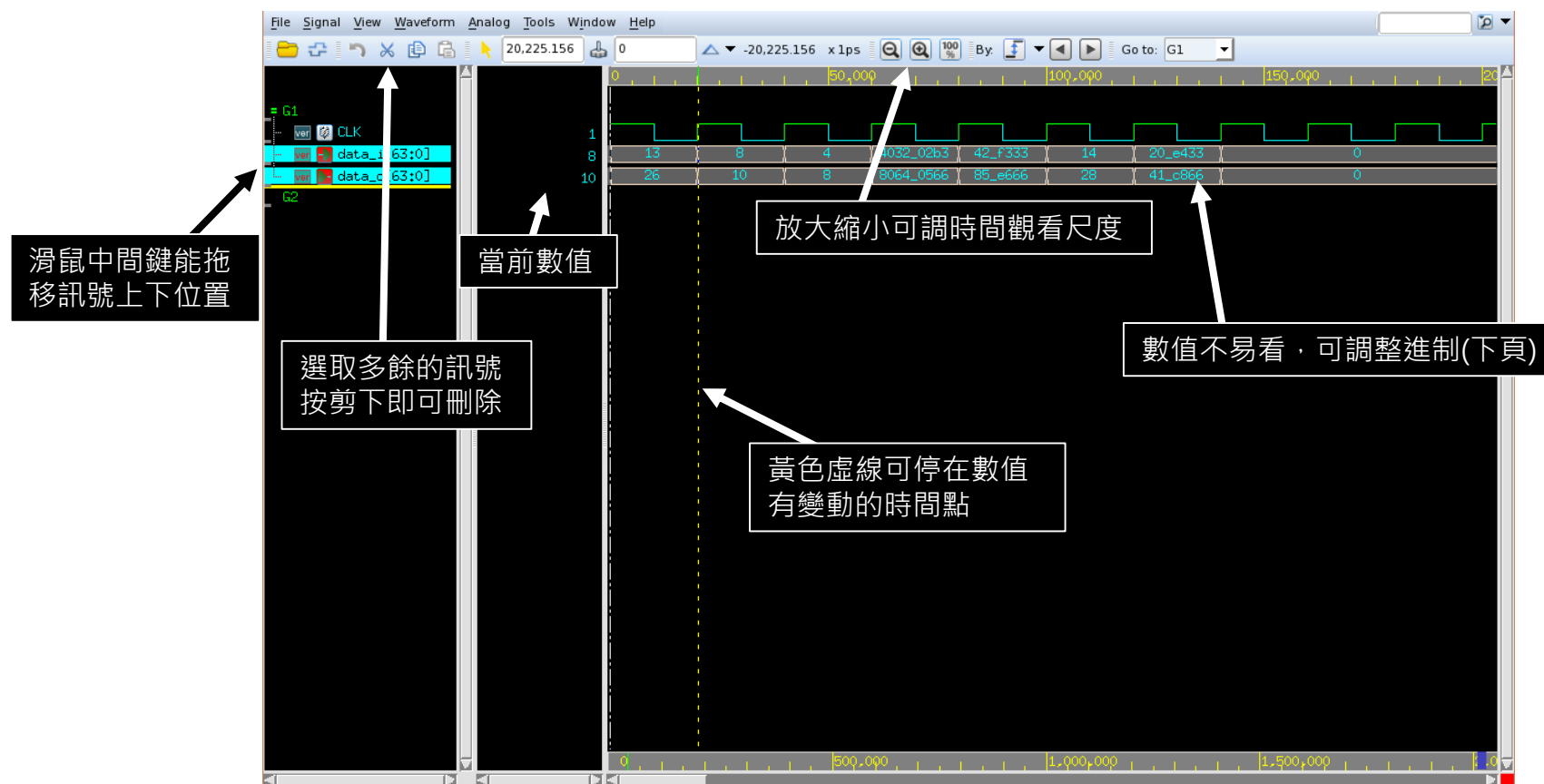
nWave使用



nWave使用



nWave使用



nWave使用

The image consists of two screenshots of the nWave software interface, illustrating the steps to change the radix of a signal.

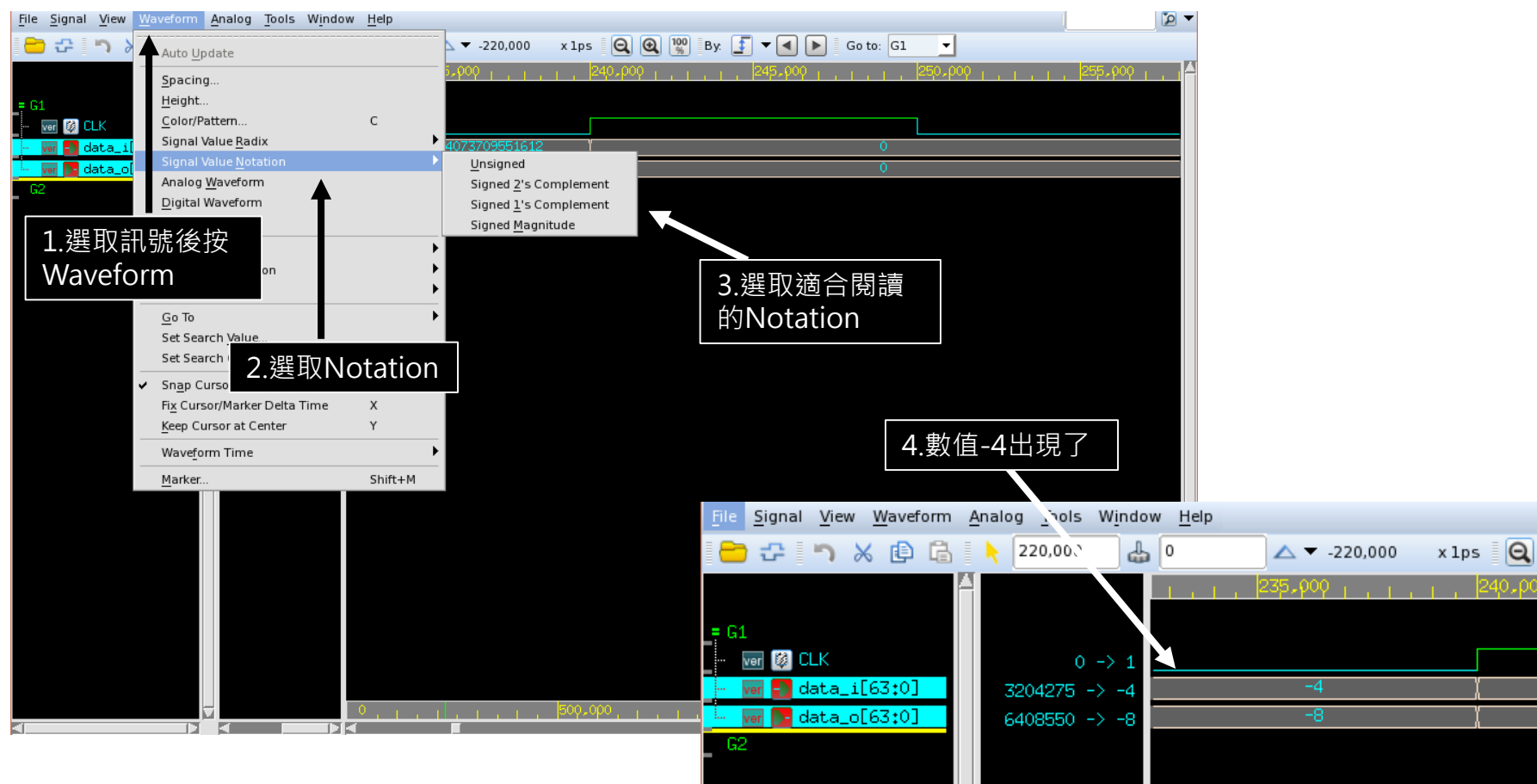
Top Screenshot: The 'Waveform' menu is open, showing options like 'Signal Value Radix', 'Signal Value Notation', 'Analog Waveform', and 'Digital Waveform'. A callout box labeled '1. 選取訊號後按 Waveform' points to the 'Waveform' menu. Another callout box labeled '2. 選取Radix' points to the 'Signal Value Radix' option. A third callout box labeled '3. 選取適合閱讀的進制' points to the 'Hexadecimal' option in the submenu.

Bottom Screenshot: The waveform is displayed with a time scale of 220,000. A callout box labeled '4. 數值-4被解讀成超大數了 需調整(下頁)' points to a value of 18446744073709551612, which is the result of interpreting the value -4 in hexadecimal.

Signal List:

Signal	Value
data_i[63:0]	18446744073709551612
data_o[63:0]	18446744073709551608

nWave使用



nWave使用

- 每次重新輸出fsdb後不用重新開nWave讀檔，只需按shift+L reload即可。
- 要先了解架構與訊號傳遞(像講義那樣知道每個指令的data flow)再用nWave確認，才能有效地找到出問題的地方